

携帯機器用生体信号処理システム

A Biosignals Processing System for Portable Devices

永島 孝浩¹、アルベルト・パラシオス²

¹ 桐蔭横浜大学大学院工学研究科、² 桐蔭横浜大学医用工学部

(2016年3月28日 受理)

あらまし 近年、回路の微細化技術の進歩は、驚異的な機能を持つ携帯機器の普及を促進している。医療の分野にしか使われなかった技術も日常で用いる携帯機器などにも使われ始めている。本論文は、生体信号の処理、携帯機器との無線でのやり取り可能なシステムについて検討を行った。本論文は、脈拍センサとCPUボードと無線用ボードで構成されたシステムのハードウェアとその一部のソフトウェアを紹介して、そのソフトウェアの開発に用いた幾つかのツールや技術などについて述べる。

1. まえがき

医療や介護の現場において、患者の測定データの集約を円滑に行うことは重要である。医療機器において有線型のものであると、使用できる場所が限られてしまう。そのため有線の装置ではなく無線機能を持つ医療機器が望ましい。ここ数年で小型化・携帯型化されスマートフォンなどの携帯機器と連携して使用する医療機器が増えてきている。しかし、このような医療機器を開発・販売している企業は機器のソフトウェアのソースは公開していないため、このような医療機器の研究・開発を行えるために、その基本的な技術の確立は必要である。

脈拍数と動脈血酸素飽和度を測定する医療機器はオキシメーターという。本論文では、

このオキシメーターの測定する項目の一つである脈拍数と似ている生命兆候である心拍数を測定して、携帯機器にそのデータを表示できるシステムについて述べる。

現在日常で使用できる生体信号を測る時計型のアクセサリは多種多様がある。それらの機器の動かす基本ソフトウェア (OS: Operating System) には、iOS、Android、Windows などがあり、機器のメーカーによって異なる。本論文で紹介するシステムの対象携帯機器の OS は iOS 対応のものである^[1]。また、携帯機器との通信には BLE (Bluetooth Low Energy) を用いる^[2]。

本システムの生体信号処理については以下の3.で述べる。本システムは有線システム^[3]から発展したため、有線システムのソフトウェアの移植について以下の4.で述べ、5.で携帯機器の GUI (Graphic User Inter-

¹ NAGASHIMA Takahiro and ² Alberto Palacios Pawlovsky

¹ Graduate School of Engineering, Tooin University of Yokohama; ² Faculty of Medical Engineering, Tooin University of Yokohama: 1614 Kurogane-cho, Aoba-ku, Yokohama 225-8503, Japan
e-mail: ¹ tm23b08h@ust.tooin.ac.jp, ² pawlovsky@tooin.ac.jp

face) と BPM (Beats Per Minute) の値表示と脈波形の描画について述べる。6. では本論文のまとめと今後残された問題について概説する。

本論文の緒論として次の 2. で使用した無線技術について簡潔に述べる。

2. 無線技術について

無線技術には、ラジオに使用する電波技術や携帯電話の無線技術や、局所無線 LAN の構築に使用する Wi-Fi などの通信技術がある。短距離無線技術には、テレビのリモコンなどに使われている赤外線通信があり、パソコンと周辺機器との接続に使用されている Bluetooth などがある。

Bluetooth とは、Ericsson Mobile Communication 社が複数の周辺機器間を相互に接続するために考案した技術である。特徴としては、周りの電波の影響を受けにくく、通信距離を数メートルから数十メートルに調整可能、遮蔽物があっても通信ができる。

Bluetooth の技術は距離が制御でき、医療機器にも既に使用されている。短所として消費電力の多さがあったが、BLE (Bluetooth Low Energy) 規格により携帯医療機器の開発また使用は促進された。

Bluetooth と BLE について

開発したシステムの無線技術は BLE である。BLE は Bluetooth のバージョンが 2009 年の 12 月に 4.0 となったときに誕生したものである。おおまかに考えると BLE は Bluetooth の低消費電力型の規格であるが、Bluetooth と異なる点がいくつかあるため、その代表的な特徴を表 1 に示す。

3. 携帯機器用の生体信号処理システム

本論文の生体信号処理システムを図 1 に示す。生体信号は脈拍センサで取得した後、Arduino 系 (UNO) の CPU (中央処理装

表 1 Bluetooth と BLE との違いについて^[4]

| | Bluetooth | BLE |
|----------|-----------------|------------|
| 周波数帯 | 2.4GHz | 2.4GHz |
| チャンネル数 | 79 | 40 |
| データ速度 | 1Mbps/3Mbps | 1Mbps |
| 最大信号速度 | 0.7Mbps/2.1Mbps | 270kbps |
| 通信距離 | 100m | 50m |
| セットアップ時間 | 6s | 0.003s |
| 消費電力 | 30mA | 0.05mA~3mA |
| 音声送信 | 可能 | 不可 |
| 最大出力 | +20dBm | +10dBm |
| ペアリング | 必須 | 不要 |
| 最大パケット数 | 1021Byte | 47Byte |

置) で AD (アナログ・デジタル) 変換など処理されて BPM が計算される。

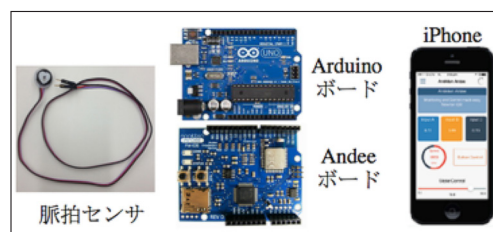


図 1 本論文の生体信号処理システム

CPU の脈拍処理用のソフトウェアは、図 2 に示す四つのプログラムで構成されている^[5]。

これらのプログラムで、USB ケーブル (有線) でパソコン上の脈拍の波形や BPM の値を表示することができる。また、Bluetooth の無線用の拡張ボードと関連のソフトウェアを用いれば、無線で携帯機器とのやり取りや、BPM 表示も可能である^[3]。

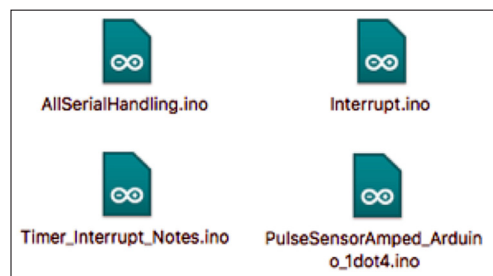


図 2 脈拍処理用プログラム群

本論文では、図 3 に示す Bluetooth の機能を搭載して、しかも CPU のプログラミン

グ言語（Arduino）で携帯機器側のアプリ（応用ソフトウェア）の機能制御とGUIの構成を可能とする Annikken 社の Andee ボードを採用している [6]。

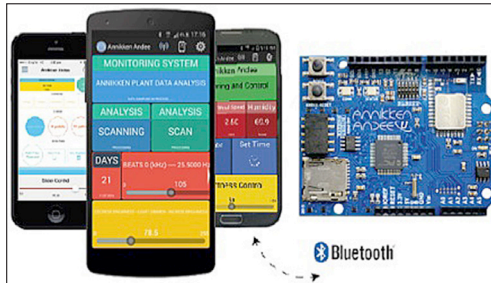


図3 Andee ボード [5]

本システムの最初の検討段階で、携帯機器側の専用の開発環境を用いることとしたが、プログラミング言語の統一化、開発をより容易にするために、Andee のボードを採用した。このボードは現在普及率の最も高い Apple 社の iOS、および Sony 社、Samsung 社などの携帯機器に用いられている Android の OS に対応している（図4）。

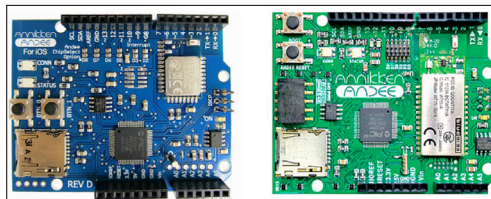


図4 (左)iOS用 Andee、(右)Android用 Andee

有線システムに用いた脈拍用のソフトウェアを Andee に使用するためにプログラムの移植を行って、携帯機器側での BPM 表示の GUI を開発した（図5）。

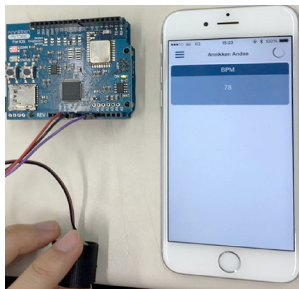


図5 BPM を表示可能なアプリの実行中の様子

4. システムのソフトウェア

本システムに、CPU の脈拍処理用プログラムの信号取得部、脈拍処理部などの四つのプログラムを Annikken 社のデモ用の表示プログラムに移植した（図6）。

CPU (4つのプログラム)



図6 移植段階の本システムのプログラム群

脈拍処理用の四つのプログラムから移植される部分や変更される部分などについて簡潔に述べる。まず、脈拍入力処理プログラム内から移植される部分について説明する。

図7の黄色の部分、CPU のボードにて、どのアナログピンで信号を取得するなどを定



図7 脈拍処理用プログラムの初期設定部分

義している。本システムでは、アナログピン A0 にて信号を取得するように定義している。また、計算された脈拍数を保持する BPM の変数、脈拍間の時間を保持する IBI (Inter Beat Interval) の変数、取得した脈拍信号 (AD 変換後) の値を保持する Signal などの変数も定義されている。

図 8 は CPU を初期設定するための `setup()` 関数の中身を示している。黄色の部分の内容は、システムのメインのプログラムに移植される。最初の二つの `pinMode` の命令で、CPU ボード上の LED (Light Emitting Diode: 発光ダイオード) を点灯や消灯をさせるためのものである。点滅は、拍動と同期して行われる。3 番目の `Serial.begin()` の命令は CPU と無線ボードとの送受信の速度を設定する。移植される 4 番目の命令は、脈拍センサの信号の観測間隔を設定する。本システムでは、その値は 2 ミリ秒となっている。

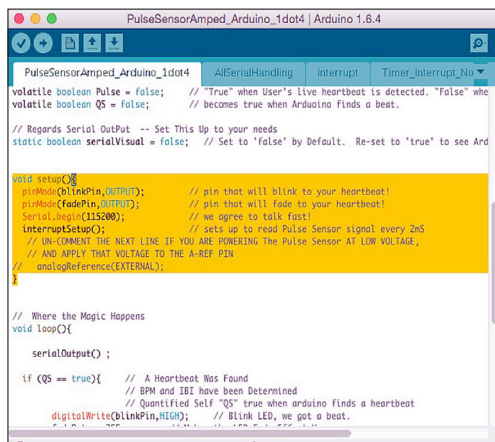


図 8 脈拍処理用プログラムの繰り返し処理の部分

脈拍処理は、図 9 の強調 (黄色の) 部分の `loop()` 関数で行われる。拍動の存在の有無を表す `QS` は `true` (真) かつデータが取得可能なら、ボード上の LED を点灯させ、フェードさせた後、脈拍処理の結果がシリアル出力に送られ、`QS` は `false` (偽) に設定される。拍動を検知できていない場合は、LED を消灯させる。20 ミリ秒の遅延時間 (`delay(20)` 命令) の後に、この一連の処理は繰

り返される。



図 9 脈拍処理用プログラムの繰り返しの部分

図 10 に信号送信用のプログラム (一部) である。このプログラムは、そのままライブラリファイルとして、携帯機器用のプログラムで使われる。その `serialOutput()` の関数では、`serialVisual` の表示制御用変数が `true` の場合は、シリアルチャンネルを使って、`Signal` の値が送信される。これは、CPU ボードはパソコンに USB (有線) で接続するとき用いる。

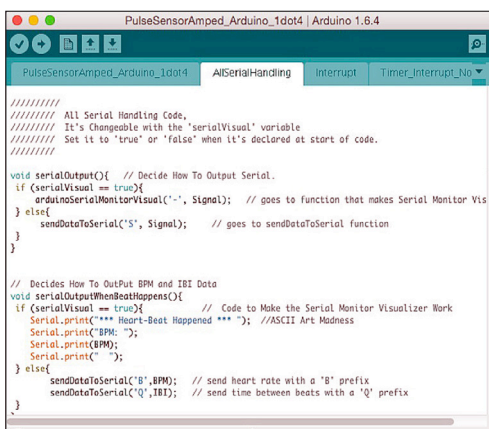


図 10 脈拍処理用プログラムの信号送信部

一方、`SerialVisual` が `false` の場合は、先頭に 'S' が付くパケット構造の `Signal` のデータが送信される。

同図のもう一つの `serialOutputWhenBeatHappens()` 関数では、`serialVisual` の表示制御用変数が `true` の場合は、表示用領域、脈拍数用の表示題名「*** Heart Beat Hap

pened ***」と「BPM: 」を出力させて、計算された BPM の値を同名の信号で送信する。serialVisual が false の場合は、シリアルチャネルに二つのパケット構造のデータが送られる。一つは B の先頭印を持って脈拍数を運び、もう一つは Q の記号で始まり、脈拍間の時間を IBI 変数の値で運ぶ。

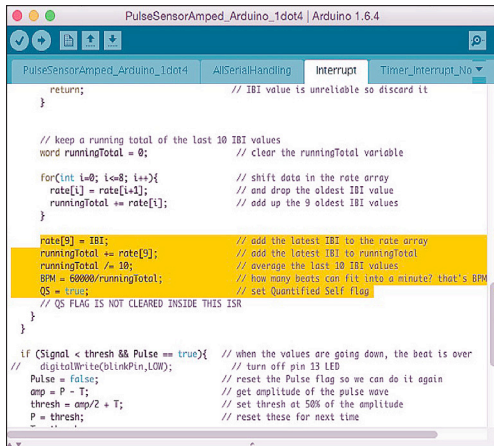


図 11 脈拍処理用プログラムの脈拍処理部

図 11 に示す interrupt のプログラムもそのままライブラリファイルとして携帯機器用のプログラムで用いられる。このプログラムには、センサから入る脈拍信号の処理の間隔を設定する interruptSetup () の関数がある。そのまま移植される 4 番目のプログラムにも図 12 に示す様に記述されている。そのプログラムには、CPU ボードによって、必要な変更など説明されている。

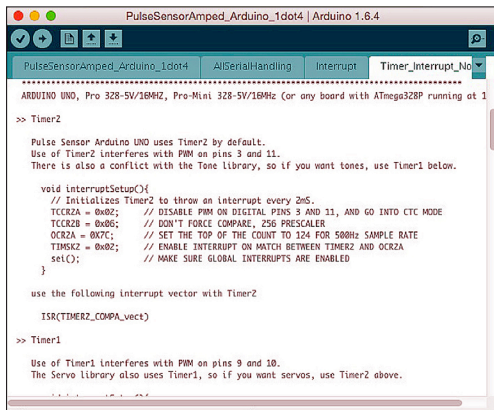


図 12 脈拍処理間隔を設定するプログラムの詳細

interruptSetup () には、CPU の出力の設定、タイマーの比較を無効にしたり、カウンターの設定をサンプリングの間隔に合わせたり、広範囲の割り込みを有効にすることが行われる。interrupt のプログラムには、上記の図 11 に強調されている部分で 10 個の IBI の時間を用いて、1 分に発生された脈拍が計算され、その値は BPM とされる。

本システムの携帯機器の表示部分 (GUI) は Arduino 言語で書かれていて、図 13 に示すデモプログラムを土台にしている。

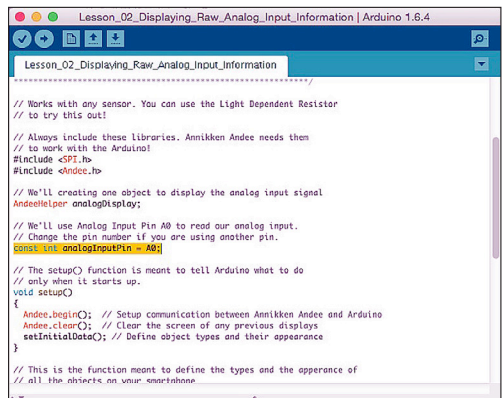


図 13 アナログデータ表示用デモプログラム

図 13 の黄色の部分にて、CPU ボードの入力 A0 をアナログデータ取得ピン (端子) と定義している。このプログラムは CPU ボードではなく、Andee のボードで実行される。初期設定に用いられる setup () 関数で Andee ボードの起動、携帯機器側のアプリの画面削除 (初期化) を行った後、同プログラム内の setInitialData () の関数を読んで、GUI を初期設定する。

Andee ボード用のソフトウェアには、AndeeClass と AndeeHelper の中核のクラスが二つある。図 13 に示されている様に analogDisplay のオブジェクトは後者のクラスと定義されている。

Andee ボード経由で設定できる GUI の各々の構成要素に識別番号が必要である。構成要素には、スライダーやボタンなどがあるため、その種類の設定も必要である。その設

定は `analogDisplay` の関数で行われる。図 14 では、一番目の `analogDisplay.setId` は携帯機器上のデータを表示する構成要素に固有の ID (identification) 番号を与えている。2 番目の `analogDisplay.setType` では構成要素をディスプレイボックスに設定している。3 番目の `analogDisplay.setLocation` ではディスプレイボックスの大きさと位置の設定をしている。4 番目の `analogDisplay.setTitle` ではディスプレイボックスに題名として使われる文字列を設定している。5 番目の `analogDisplay.setData` ではディスプレイボックスの表示用の初期値を設定している。この五つの命令は、何の構成の設定にも用いられる。

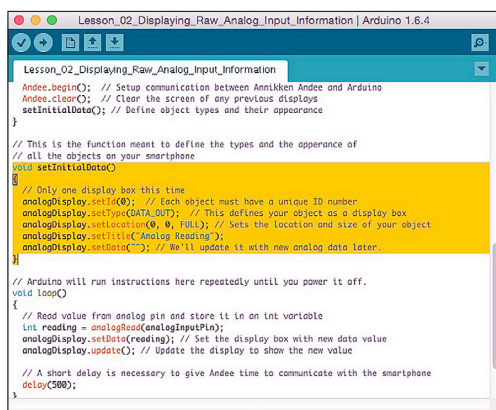


図 14 データ表示部 (GUI) の設定部分

図 15 の黄色部分は、GUI 関連で `loop()` 関数で繰り返して実行される命令である。土

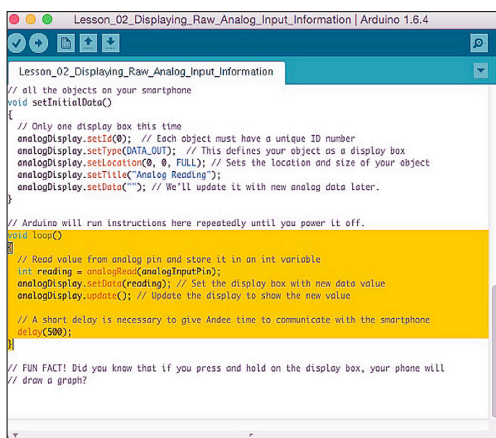


図 15 GUI に表示させるデータに関する命令

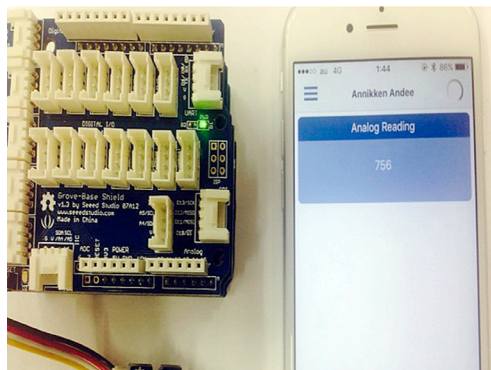


図 16 携帯機器上にアナログデータ表示の様子

台のプログラムでは、CPU ボードの A0 の入力で読み取ったデータを整数型の `reading` 変数で `analogDisplay.setData` でディスプレイボックスに表示するように設定されている。3 番目の `analogDisplay.update` でディスプレイボックスの表示は更新される。500 ミリ秒の待機時間後この処理は再び繰り返される。可変抵抗で分圧された電圧の AD 変換後の値を示す応用例の出力を図 16 に示す。

この応用例には Seeed Studio 社の拡張ボード^[7]は使われている。本システムのソフトウェアでは、上記に説明した命令の変更が必要である。変更部分について下記に述べる。

図 17 の黄色の部分において開発したプログラムでは、経由用の変数 (図 15 の `reading`) は不要で、入力信号ではなく GUI で示

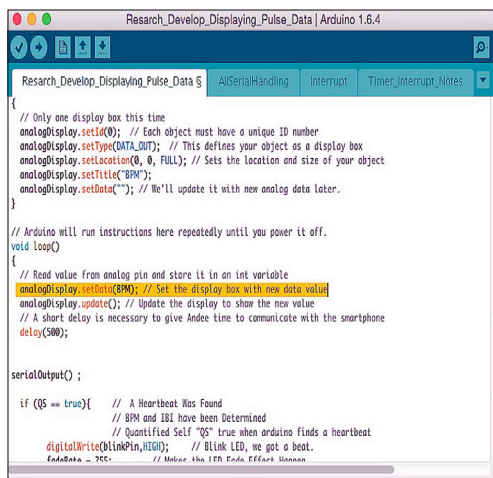


図 17 本システムの GUI 用の設定部分

される値 (BPM) は使用される。

同図には、図 8 の CPU のプログラムから移植された命令の一部も確認できる。

本システムでは、GUI に BPM と脈拍のデータを両方同時に表示させるために図 18 の上部 (最初の五つの命令) に示す様に GUI にもう一つの構成要素 (識別番号 1) は追加されている。

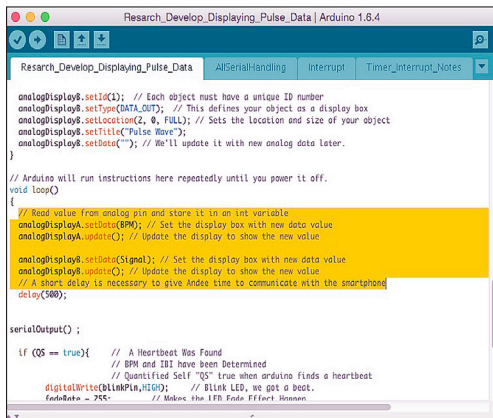


図 18 二つの GUI に表示させるデータの設定部分

各々の要素を制御するために、本ソフトウェアでは二つのオブジェクト `analogDisplayA` と `analogDisplayB` を用いている。

図 18 にて `analogDisplayA.setData(BPM)` と `analogDisplayB.setData(Signal)` で BPM の値と脈拍のアナログデータを表示させるよう設定されている。

5. BPM と脈拍信号の同時表示

上記に説明した様に、システムを構成して、本ソフトウェアを実行させると、図 19 の様に、携帯機器の GUI に BPM と脈拍信号の (数値化した) 値は同時表示される。図 20 の BPM の値は 78 のため正常に脈拍数が計測できたと言える^[8]。

Andee の携帯側の新型のアプリにより、CPU から送られるデータは携帯機器に一時的に記憶されるため、そのデータをグラフ化できる。表示枠とグラフは同時に表示できないが、ボタンを押して、グラフに切り替え表

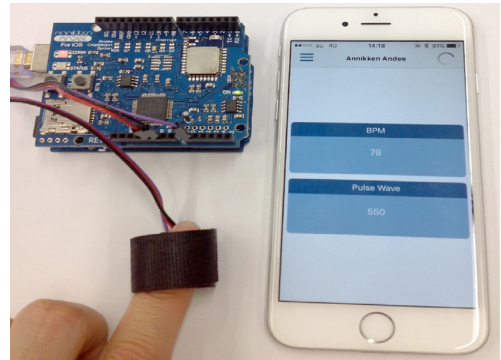


図 19 BPM と脈拍信号の同時表示の様子

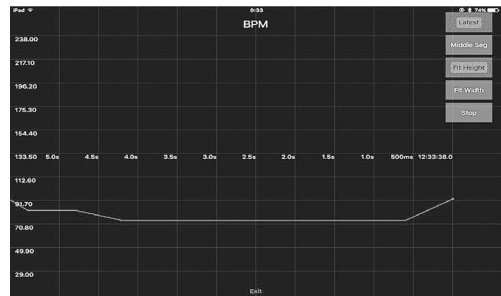


図 20 BPM を iPad にてグラフとして表示させた図

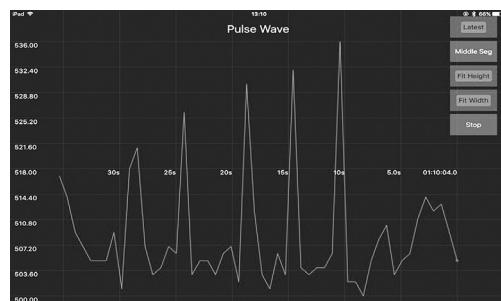


図 21 脈拍を iPad にてグラフとして表示させた図

示が可能である。なお、本論文の執筆現在にはアプリ側の不具合により iOS の iPhone では表示できなく iPad のみで表示可能となっている (図 20 と図 21)。

6. まとめと今後の課題

本システムでは、開発したソフトウェアを用いて脈拍センサにて取得した信号を Arduino 系の CPU ボードと Andee ボードの組み合わせで脈拍信号の処理を行い、BLE を通して携帯機器 (iPad) 上に BPM と脈拍を数

値としては両方同時に表示が可能である。これらのデータに対応するグラフも別々に表示可能である。

palmic_j.html



図 22 既成の無線型パルスオキシメーター Palmic^[9]

図 22 に示す様に、市販の Palmic のオキシメーターは、動脈血酸素飽和度 (SpO₂) と脈拍数を測定し、無線にて動脈血酸素飽和度、脈拍数、脈波形を iPhone や iPad へ送信し専用のアプリを介してそれぞれを数値と波形は表示可能である。現在 Andee のアプリで独立したグラフ表示ができないため、携帯機器上の GUI の画面を切り替えずにそのままデータ表示部にグラフを表示させることは今後の課題として残される。

【参考文献】

- [1] <http://www.apple.com/ios/>
- [2] <https://www.bluetooth.org/ja-jp>
- [3] (私報) 永島孝浩、無線機能を持つパルスオキシメーターの研究・開発、桐蔭横浜大学、医用工学部、臨床工学科、平成 25 年度卒業論文
- [4] <http://www.musen-module.com/news/2014/nov2014/news01.html>
- [5] [https://github.com/WorldFamousElectronics/Pulse Sensor _Amped_Arduino](https://github.com/WorldFamousElectronics/Pulse-Sensor_Amped_Arduino)
- [6] www.annikken.com
- [7] www.seeedstudio.com
- [8] ME の基礎知識と安全管理 28 頁、日本生医工学会 ME 技術教育委員会 (監修)、南江堂 (出版)
- [9] <http://www.micdent.com/page/>